NOVA IMS Information Management School

MGI

Mestrado em Gestão de Informação
Master Program in Information Management

Using LUCAS survey and Recurrent Neural Networks to produce LCLU classification based on a Satellite Image time series of Sentinel-2

Nuno Alexandre Pereira da Silva

Dissertation presented as partial requirement for obtaining the Master's degree in Information Management

NOVA Information Management School
Instituto Superior de Estatística e Gestão de Informação
Universidade Nova de Lisboa

NOVA Information Management School

Instituto Superior de Estatística e Gestão de Informação

Universidade Nova de Lisboa

USING LUCAS SURVEY AND RECURRENT NEURAL NETWORKS TO PRODUCE LCLU CLASSIFICATION BASED ON A SATELLITE IMAGE TIME SERIES OF SENTINEL-2

by

Nuno Alexandre Pereira da Silva

Dissertation presented as the partial requirement for obtaining a Master's degree in Information Management, specialization in Knowledge Management and Business Intelligence

Advisor / Co-supervisor: Mário Sílvio Rochinha de Andrade Caetano

Co Advisor*:* Mauro Castelli

February 2021

ACKNOWLEDGEMENTS

I would like to thank both my coordinators, Mário Caetano and Mauro Castelli, for all the help and time invested on this dissertation. I would also like to thank Direção-Geral do Território (DGT) for the support throughout this study, specially to Hugo Costa and Pedro Benevides. Additionally, I would like to thank the project foRESTER (PCIF/SSI/0102/2017), as part of this study was done in the scope of this project.

ABSTRACT

The need of timely and accurate information for the territory has increased over the years, making Land Cover Land Use (LCLU) mapping one of the most common application of remote sensing. Recently, the advances in satellite technology and the open access policies for remote sensing data increased the interest in exploring satellite image time series. In addition, the attention of researchers has shifted from standard machine learning algorithms (e.g., Support Vector Machines and Random Forest) to Recurrent Neural Networks due to their ability of exploiting sequential information. However, acquiring reference data to train these algorithms is still a hurdle. This study aims to evaluate the capability of a Gated Recurrent Unit in performing pixel-level LCLU classification of a satellite image time series, using Sentinel-2 imagery and having the LUCAS survey as reference data. To assess the performance of our model we compared it to state-of-the-art classifiers (SVM and RF). Due to the unbalance nature of the LUCAS survey, we applied oversampling to this dataset to increase the performance of our models, testing three different oversampling techniques. The results attained showed that Recurrent Neural Networks did not outperform the other state-of-the-art algorithms, when trained with a limited number of sampling units, and that oversampling the LUCAS survey increased the performance of all the classifiers. Finally, we were able to demonstrate that it is possible to produce LCLU classification of satellite image time series using only open-source data by using Sentinel-2 imagery and the LUCAS survey as refence data.

KEYWORDS

LCLU classification; LUCAS survey; Recurrent Neural Networks; Oversampling; Sentinel-2

INDEX

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS AND ACRONYMS

LCLU	Land Cover Land Use
MMU	Maximum Mapping Unit
RF	Random Forest
SVM	Support Vector Machine
CNN	Convolutional Neural Network
AE	Autoencoder
RNN	Recurrent Neural Network
ESA	European Space Agency
LSTM	Long-Short Term Memory
GRU	Gated Recurrent Unit
CLC	CORINE Land Cover
LUCAS	Land Use and Land Cover Area Survey
SMOTE	Synthetic Minority Over-Sampling Technique
DGT	Direção-Geral do Territótio
MSI	Multispectral Imager

1. INTRODUCTION

The importance of Land Cover Land Use (LCLU) mapping is increasing over time due to the need of having timely and accurate information about the range and nature of the land resources. Having this in mind, generating LCLU maps is one of the most frequent applications of remote sensing (Storie & Henry, 2018). The purpose of LCLU classification is to describe the land surface information (type and use), by assigning to each minimum mapping unit (MMU) one of a predefined set of labels (Helber et al., 2019). Information regarding LCLU is crucial for many geospatial applications (Zhang et al., 2019). For urban environments, LCLU maps are critical for the study of city planning, house rents, urban transportation network and other phenomena like urban heat island effects (Huang et al., 2018). For environment management, it is considered one of the essential climate variables, as it is crucial to monitor climate change effects, to aide in disaster prevention and to manage natural resources (Pelletier et al., 2019). LCLU information has many other applications, like the one presented in Ho Tong Minh et al. (2018), where it was used to monitor the pollution of drinking water by nitrates due to the intensive use of agricultural fertilizers. This was achieved through the mapping of the winter vegetation coverage present in the study area, which could help understand the amount of absorption of those nitrates by the soil.

Since the manual production of LCLU maps through visual interpretation of the satellite images is costly in time and labour, especially when covering large areas (Pflugmacher et al., 2019), many studies have focused their attention on automating this process by using a machine learning algorithm to classify remote sensing images (Khatami et al., 2016). Standard machine learning algorithms like Random Forest (RF) and Support Vector Machines (SVM) have been widely used by the remote sensing community to perform this type of tasks (Pelletier et al., 2016). However, in more recent years, researchers have turned their attention to Deep Leaning, as this type of models have found a lot of success in many other applications like speech recognition and computer vision (Ma et al., 2019). Huang et al. (2018) found a lot of success when applying a semi-transfer Convolutional Neural Network (CNN) to make land use classification in an urban environment, which incorporated an already trained deep CNN (AlexNet) and a shallower CNN that was trained with multispectral data. Chen et al. (2014) was able to extract high level and abstract features from satellite images through the application of a deep architecture of autoencoders (AE), which was able to handle variations present in remote sensing data, like sensor rotation and atmospheric conditions. Mou et al. (2017) used a Recurrent Neural Network (RNN) to exploit the temporal correlation present in remote sensing images, which is not leveraged when using other classifiers like RF, SVM and CNN, that consider each pixel as an order less data point, therefore ignoring the temporal sequence inherent to satellite images.

The usage of time series, as exploited in the study referred above, can be helpful to distinguish land cover classes that have different temporal behaviours. However, the analyse of multi-temporal remote sensing data is still a challenge (Ienco et al., 2017), as most of the LCLU classification studies centre on the spectral and spatial domain of the satellite images (Rußwurm & Körner, 2018). More recently, the interest in remote sensing time series has risen due to the availability of more satellite data, that was boosted not only by the launch of new satellites with a small revisit time, but also by the open policies which make this data accessible at no cost (Flamary et al., 2015). The Sentinel-2 mission from the European Space Agency (ESA) is a good example of this, as it supplies multispectral

high-resolution satellite imagery from around the globe with a 5 day revisit time (Pahlevan et al., 2017). To properly exploit this sequential information, the RNNs have been used in most studies, especially regarding two types of networks, the Long-Short Term Memory (LSTM) and the Gated Recurrent Unit (GRU), as these units can model long term temporal relationships due to their gated structure (Pelletier et al., 2019).

Supervised learning methods have been considered more suitable and have found more success in LCLU classification among the remote sensing community. However, these types of algorithms require a set of pre-classified data to define their parameters. This training process is very influenced by the amount of sampling units available alongside with their label accuracy (Flamary et al., 2015). There are some different ways to get this reference data to then feed to the algorithms: it can be done by a field survey, as performed in Ho Tong Minh et al. (2018), but this is very expensive and time consuming, and therefore only feasible for smaller samples; or by using an already existing reference map, which can embrace small to regional areas, like a country in Huang et al. (2018), or it can encompass an wider area, like in Zhang et al. (2019), which used CORINE Land Cover (CLC), an European data inventory produced by the European Union Copernicus program. However, this may lead to some decrease of the label quality of the training data if the date of the reference map is significantly prior to the date of the satellite images, as some changes in the ground may have occurred (Zhang et al., 2019). It is also important to notice the reference map proprieties, including the size of the minimum mapping unit (MMU), the underlying percentage of error (if existing), and some generalizations that could have been made, which can affect the quality of the labels. This could be seen in CLC, that has a MMU of 25 hectares (ha), meaning that areas smaller than the specified mapping unit are going to be encompassed by the surrounding areas to meet the MMU (Agency, 2020). More recently, some studies have used the Land Use and Land Cover Area Survey (LUCAS) as their reference dataset, to train algorithms for LCLU classification in Europe, even though it is originally produced with the intent of statistical estimation (Douzas et al., 2019). This dataset is produced on a three-year basis since 2006 by the Eurostat, the statistical office of the European Commission. This survey gathers LCLU information, as well as environmental characteristics for all sample points which are spread on the entire European Union territory in a 2 km regular grid (Mack et al., 2017). This information is collected not only by photointerpretation but also though field visits conducted by trained experts, which ensures higher level of accuracy and consistency on the information assign to each data point (Weigand et al., 2020).

In this study, we will assess the performance of a Recurrent Neural Network model to make a pixel-level LCLU classification of a satellite image time series, having the LUCAS survey as the reference data. In more detail, we will compare the performance of a Gated Recurrent Unit against two other state-of-the-art algorithms (Random Forest and Support Vector Machines) to perform a LCLU classification of a yearlong time series. To train these models, we will use information derived from satellite image composites, one for each month, sourced from Sentinel-2 imagery, having as reference data a filtered version of all LUCAS sample points located in Continental Portugal. In addition, we will run three oversampling techniques (i.e., SMOTE, Borderline SMOTE and Geometric SMOTE) to reduce the imbalance problem present in the training data, due to the disparity of the number of sampling units existent for each LCLU class. The data preparation and the methodological approaches used in this thesis were performed within the General Directorate for Territory (*Direção-Geral do Território*, DGT) program to develop operational methodologies for supervised classification of LCLU at national level. In particular, the Sentinel-2 intra-annual surface reflectance imagery was

provided by DGT, as well as the validation dataset with ground-truth obtained from label interpretation at DGT. Unlike other approaches made within this DGT's program, that combine LUCAS survey with other sources to create the reference data that will train the algorithms, we decided to only use LUCAS as our reference data, which will not only make possible the comparison with other studies that only use LUCAS as their training data but will also prove the potential of using the LUCAS survey for LCLU classification. It is important to notice, that this decision will lead us to have a limited training sample, which may injure the performance of the GRU, as this type of model usually requires a considerable number of sampling units to be trained properly.

This document is organized in the six following sections: Section 2 contains the literature review that served as basis for this study, Section 3 encompasses the methodology followed, Section 4 displays the results of the study, Section 5 presents the conclusions taken, and Section 6 contains the limitations faced during this study and the recommendations for future works.

2. LITERATURE REVIEW

In this section, we present the literature review that served as basis to our study, which is organized in two main parts. The first will be focused on the machine learning algorithms used in our study and the second sub section will be centered around the LCLU classification methodologies.

2.1. LEARNING ALGORITHMS

2.1.1. Random Forest

The Random Forest (RF) classifier is a state-of-the-art machine learning algorithm for remote sensing image classification, as this method produces high classification accuracies, like other more complex algorithms, but with a lower computational cost and being very stable regarding the choice of parameters (Inglada et al., 2017). In addition, it can be trained in a high dimensional dataset without having a considerable overfitting and is somewhat robust to outliers and noise present in the data. The RF algorithm consists of a big number of decision trees, which are constructed with the training data, forming an ensemble classifier (Heine et al., 2016). Each tree is built based on a bootstrap sample of the training dataset and each split node is defined by choosing the best split among a random subset of variables, instead of choosing the best split among all variables, which adds an additional layer of randomness to the bagging method (Liaw & Weiner, 2002). Despite of the fact that this approach creates weaker individual trees, it will reduce the correlation between them, which in the end, will increase the overall generalization power of the classifier (Pelletier et al., 2017). Mack et al. (2017) used RF to make LCLU classification on Landsat data. In this paper, the authors applied the RF twice, being the first time to identify the most uncertain areas in the training region, and afterwards, manually generating more data points from these uncertain areas and including them into the dataset to train the final classifier. Weigand et al. 2020) choose the RF to test the efficiency of four different pre-processing schemes to make LCLU classification based on LUCAS survey. To eliminate the randomness inherently present in the RF algorithm, the authors ran each model 100 times to properly evaluate each different approach.

2.1.2. Support Vector Machine

Support Vector Machines (SVM) have also drawn the attention of the remote sensing community, due to their generalization ability and their capability of working well with a low number of training data points (Mountrakis et al., 2011). SVM seeks to find an optimal hyperplane that completely separates the training points of each class. To find this hyperplane, the algorithm only considers the training data points close to the class boundary, that are called support vectors, therefore, working well with small samples (Fauvel et al., 2008). This optimal hyperplane not only aims to split the classes, but also to maximize the margin, that is the distance between the support vectors and the hyperplane. The size of the margin is a key metric for the generalization power of the SVM (higher the margin, higher the expected generalization) (Melgani & Bruzzone, 2004). However, the SVM assumes that the problem is linearly separable, which is most of the times not the case. To solve this, techniques like the soft margin method and the kernel trick can be used (Mountrakis et al., 2011). The soft margin technique is the inclusion of a penalty parameter that will allow the SVM to have some misclassified data points when defining the optimal hyperplane (Zhuo et al., 2008). The kernel

trick consists of applying a kernel function that projects the training data into a higher dimensional feature space, which would make easier to find a linear separation between the classes (Xia et al., 2015). Taati et al. (2015) found success when using SVM to generate a land use map, having Landsat 5 images as training data. With this approach, the authors were able to increase the overall accuracy of the map in more than 6%, when compared to their baseline. Deilman et al. (2014) was also able to achieve better results when applying SVM for land cover classification, as this algorithm was able to address the issue of the mixed pixels present in the satellite image.

2.1.3. Neural Networks

Neural networks are the basis of Deep Learning and consists of a loosely mathematical abstraction that attempts to mimic the learning process of the brain (Storie & Henry, 2018). These networks are composed by layers of neurons, also called units, that are connected by weights, which define the relationship between the neurons. At the end of each neuron, an activation function is applied (e.g., ReLU, sigmoid) before the value goes to the next neuron. During the training process, not only the weights are defined, but also the bias of each neuron, with the goal of learning the underlying patterns in the data. These parameters are obtained through backpropagation of the error during the training process. This process uses a loss function which aims to reduce the difference between the value predicted by the network and the proper output (Zhang et al., 2019). Neural networks are composed by an input layer, that introduces the points to the network and an output layer that exports the predicted results of the network, and in between those two layers there are "hidden" layers. The fact that a neural network contains multiple "hidden" layers is what makes the network "deep" and therefore, is considered "Deep Learning" (Litjens et al., 2017).

2.1.3.1. Recurrent Neural Networks

Recurrent Neural Networks are a simple adaptation of a standard feedforward neural network that enables the modelling of temporal dependencies (Sutskever et al., 2011). This is possible because the network feeds itself past information, since the output of the neuron at time t-1 is going to be reintroduced into the neuron alongside with the next input at time t (Ienco et al., 2017). Taking this into consideration, RNN can take advantage of the sequence-based structure present in remote sensing images, otherwise not harnessed by other algorithms commonly used to deal with this type of data (e.g., SVM, RF, CNN). These methods consider each pixel as an order less data point, which means that no spectral correlation nor band-to-band variability is taken into consideration (Mou et al., 2017). Nevertheless, standard RNN's fail to learn if the sequence of the data is greater than 5 – 10 time steps, due to the issue of vanishing and exploding gradients. This happens because the temporal evolution of the backpropagated error exponentially depends on the size of the weights, so it may lead to a big fluctuation on the weights (exploding gradients) or a severely slow learning process (vanishing gradients), either way failing to converge the weights (Gers et al., 1999); (Hochreiter & Schmidhuber, 1997).

To solve this problem, a new type of RNN called LSTM was proposed in Hochreiter & Schmidhuber (1997), to learn long term dependencies. To accomplish it, the LSTM neuron is composed by two cell states (the memory and the hidden state) and three gates (the forget gate, the input gate, and the output gate). This gated structure is used to deal with the vanishing/exploding gradients problem and to control the amount of information that is kept or forgotten during the process. These gates are implemented by sigmoid functions, that can range between 0 and 1, where 0 means that the gate is

"fully closed" and 1 means that the gate is "fully opened" (Ienco et al., 2017). The forget gate decides how much information should be discarded from the cell's internal state. The input gate determines the adequate amount of information that should be kept from the new input, considering that not all of it can be useful. The output gate defines how much to filter from the cell state to be outputted (Lyu et al., 2016). The LSTM has been able to find success in a variety of studies related with remote sensing, like in Lyu et al. (2016), where it was able to learn a change rule in his core memory cell. The model proposed was not only capable to identify if a pixel had changed (binary change), but it was also able to identify the type of change (multi-class change). In Ienco et al. (2017), the LSTM was not only applied as a classifier, but it was also used to create a new representation of the multi-temporal data to latter feed another machine learning algorithm. In the end, the modelling of the temporal correlation of the remote sensing data proved to improve the results of the study, especially on highly mixed land cover classes and classes with low representation.

In the literature we also find another type of RNN applied to remote sensing, the Gated Recurrent Unit (GRU). It follows a similar philosophy as the LSTM, since it also contains gates and two cell states (the memory and the hidden state), however it only has two gates (update and reset gate) instead of three, which makes this model simpler to implement (Ho Tong Minh et al., 2018). This means that the GRU will have less parameters to learn than the LSTM, making this model easier to train with a limited number of data points and less prone to overfit (Mou et al., 2017). GRU units that are more susceptible to capture short-term dependencies will tend to have the reset gate more "open", since this gate controls the amount of information from the previous timestamps that should be integrated with current information. On the other hand, if the unit is more prone to capture long term dependencies, it will have the update gate more "open", as this gate controls the amount of information retained in the current hidden state from the previous hidden states and from the current timestamp, acting similarly to the memory cell in the LSTM unit (Ho Tong Minh et al., 2018). This type of RNN proved to be very effective in land cover classification of hyperspectral images in Mou et al. (2017) and in land cover classification of Radar images in Ho Tong Minh et al. (2018). In both studies, the GRU reached better results than the LSTM.

2.2. LCLU Mapping with Satellite Images

There are two main methods to produce LCLU maps: a manual way, that consists of photointerpretation by the human eye, which is not appropriate for operational LCLU mapping of large areas, as well as being very expensive in terms of time and resources; and an automatic method, that uses remote sensing images and classification algorithms to automatically generate the LCLU maps (Douzas et al., 2019). The automation of the production process of LCLU maps can be valuable for many stakeholders, as the standardization of this process can improve over time change comparisons (Storie & Henry, 2018). However, for this type of task, there is a preference for supervised classification algorithms, which require labelled training samples (Pelletier et al., 2019). This prerequisite is one of the biggest hurdles to overcome in the automation of the production of LCLU maps, as labelled sampling units are "hard to get" and may not be enough to properly train the algorithms (Pan et al., 2017).

The acquisition of labelled data can be achieved through different ways. One of them, is based on field survey, or visual interpretation of orthophotos or very high spatial resolution satellite images.

This approach is very time and labour costly since it consists of going directly to the area of the data point and manually register the required information, or having technicians analysing and labelling each MMU (pixel or polygon) present in one or multiple orthophotos, respectively. In Ho Tong Minh et al. (2018), a field survey was conducted to attain the land cover class and the amount of vegetation of each of the 194 sample points used to conduct the study. However, as this method of extracting information is very expensive (timewise and moneywise), it may explain the low number of labelled sampling units used in the study and the inaptitude of this method to be used in a study that needs more pre-classified data. The analysis of orthophotos and very high spatial resolution satellite images is widely used by DGT. Their projects/studies usually combine this type of data collection with the data of already existing maps to get the reference data needed to train their machine learning algorithms.

Another way to get labelled training data, is through existing LCLU data, as in Storie & Henry (2018), where the municipality of the study area, provided the maps of the region to the authors, enabling them to attribute the proper LCLU class to each Landsat data point. In Pelletier et al. (2019), a reference map, alongside with a farmer´s declaration and a field survey, were used to collect the land cover information needed to train a temporal convolutional neural network, with the goal of learning the temporal and spectral features present in a time series of satellite images. The use of previously made maps as reference data for remote sensing studies, can present some inaccuracies, if the date of the reference map differs from the date of the satellite images used, as in the meantime, some changes to the ground truth may have occurred. This was the case in Huang et al. (2018), where a land use map was done for Hong Kong and Shenzhen. In the case of Hong Kong, the land use reference data was from 2013 and the satellite images were from 2015. In this case, the authors decided to use the old reference data anyway, since there were hardly any changes in the land use boundaries.

A similar alternative is using already build datasets available on the Internet, as in Chen et al. (2014) that used two hyperspectral datasets to validate their model that combines a Principal Component Analysis, a stacked autoencoder and a logistic regression to make land cover classification. This method aimed to extract deep and abstract features to be able to deal with variations that can be found in remote sensing images, like sensor rotations and atmospheric conditions. Pan et al. (2017) used three hyperspectral datasets to also create a model to perform land cover classification. To handle the lack of labelled training pixels and to properly train the deep learning model, the authors took advantage of an unlabelled neighbourhood of pixels that surrounded the labelled pixels. Helber et al. (2019) developed a new free and publicly available multi-spectral dataset for deep learning applications. This new dataset was created with the goal of making a large amount of remotely sensed data openly accessible for commercial and non-commercial applications to boost the innovation in this field. The multi-spectral images used to create this dataset were from Sentinel-2A, one of the two satellites of the Sentinel's constellation, as they are openly and freely provided by the EU Copernicus program.

This European program also produces a dataset that is used as reference data for LCLU problems as well, CORINE Land Cover (CLC). This data inventory maps the European territory in 44 land cover classes, with a minimum mapping unit (MMU) of 25 hectares (ha) and a minimum width of 100m. It is created through visual interpretation of high-resolution satellite imagery by most countries, where others apply semi-automatic methods. It was firstly produced having as the reference year 1990, and

it took 10 years to produce. The following updates took place in 2000, 2006, 2012 and more recently in 2018. Along the way, the production time of CLC has been reduced, taking only one year and half to produce the most recent version (2018) (Agency, 2020). It has been used as reference data in studies like Ienco et al. (2017), where the authors used the 2012's version of CLC alongside with the farmer´s graphical land parcel registration of 2014 (RPG), to create one of their reference datasets used to model the temporal behaviour of different land cover classes. Zhang et al. (2019) also used CORINE, as well as Urban Atlas, to define their land cover classes with the objective of training part of their iterative model, that aimed to simultaneously classify an image with land cover and land use classes. This iterative model had the particularity of using the LC probabilities produced in the first iteration, in the prediction of the LU probabilities, which subsequently, would feed the LC prediction of the next iteration.

The European Union produces another dataset that is also used as reference data for land cover classification studies, the Land Use and Coverage Area frame Survey (LUCAS). This survey is conducted by Eurostat on a three-year basis since 2006, being the latest version carried out in 2018. The points are recorded based on a 2 km regular grid across the territory of all 28 EU countries (Leinenkugel et al., 2019). LUCAS is an in-situ survey, as the sample points are in a first instance photo interpreted and then, a subsample of them, are visited by the surveyors to record the land cover and land use class of that sample point in the field (Griffiths et al., 2019). During the field visits, the surveyors try to get as close as possible to the theoretical location of the LUCAS point to collect the information needed. However, due to GPS positional errors, there can be an offset between the theoretical LUCAS point and the point where the information is recorded. Despite this, the data collected in these field trips will always be linked to the theoretical location of the point (Weigand et al., 2020). The LCLU class assigned to the point is relative to the 1.5 m radius circle around the theoretical location, except for heterogeneous classes, where the circle radius is extended to 20 m (Mack et al., 2017).

LUCAS takes into consideration 84 different land cover subclasses, which belong to one of the following major classes: Artificial Land (A), Cropland (B), Woodland (C), Shrubland (D), Grassland (E), Bare soil, Moss and Lichens (F), Water (G) and Wetlands (H) (Leinenkugel et al., 2019). The main goal of LUCAS is to produce statistical estimation and to provide information regarding changes in management and coverage of the EU territory to decision makers and to the public. However, LUCAS can also be successfully used to train and test machine learning algorithms to solve remote sensing problems (Pflugmacher et al., 2019). Some different approaches were taken regarding the use of LUCAS database in classification problems.

In Leinenkugel et al. (2019), the LUCAS survey points were used in combination with other open-source repositories of geodata (CLC, Natura 2000, Riparian Zones, Urban Atlas and OpenStreetMap), to train a LCLU classifier. In this study, only a part of the LUCAS sampling units was used in the training of the algorithm, as the LUCAS survey was chosen as the main data source for the validation set. For some classes, there were not enough data points in LUCAS database alone, therefore, the other data sources were used to complete the validation set for those classes. Griffiths et al. (2019) also used LUCAS in combination with another dataset to train the proposed model. In this study, the LUCAS points were only used as reference for the classes of urban, forest and water. For all the other land cover classes the reference data came from GSAA (Geospatial Aid Application). Another study that used LUCAS in combination with other dataset was Close et al. (2018). In this study, the LUCAS

survey was only merged with other data source (in this case the Belgium inventory grid), to produce the test set, in contrast with the two other studies mentioned above, that combined LUCAS and other sources for both training and validation datasets.

Other studies like Mack et al. (2017) used LUCAS alone as their reference data for both training and testing of their proposed semi-automated LCLU classification model. Weigand et al. (2020) also used LUCAS survey alone as their reference data. In this study, a comparison between four different methods was conducted to produce a LCLU classification model. The different approaches vary regarding the filters used to select the LUCAS sample points to include in the training and validation datasets and the combination of land cover classes chosen to classify each data point. In contrast with the other studies discussed above, Pflugmacher et al. (2019) used LUCAS survey in his full scope, meaning that instead of using it to train a model to make LCLU classification in a specific area/ country, it aimed to make a LCLU mapping for the entire European territory, even for countries not belonging to the European Union, there for not present in the LUCAS survey.

Using the LUCAS survey as the reference data to train a supervised classifier can be difficult because of the asymmetry of the number of points per LCLU class, which results in an imbalanced learning problem. This is particularly prone to happen in LUCAS survey, due to the adopted sampling strategy (2 km regular grid over the territory) (Douzas et al., 2019). Training an algorithm with an uneven dataset makes the model strongly biased towards the most represented class, usually attaining poor results on the minority class (Sáez et al., 2016). This problem has been researched over the past years with a special emphasis on binary classification, however, the issue is much more concerning when it happens in a multi classification scenario, as the relationship between classes is not clear (Krawczyk et al., 2018). This issue can be handled with different methodologies (e.g., data-level, algorithm-level, or ensemble methods). The data approach is the most popular strategy in the remote sensing community (Douzas et al., 2019). One application of this type of strategy was done in Leinenkugel et al., (2019), where the sampling units of the most representative classes were randomly reduced until it reached the median value of all the classes. For the less representative classes, the data points were increased by randomly adding training sample points from other data sources. One of the most popular data modification techniques is the Synthetic Minority Over-Sampling Technique (SMOTE), which consists of the creation of synthetic data points alongside the lines that connect the k nearest neighbours on the minority class (Fernández-Navarro et al., 2011). In Douzas et al. (2019), a comparison between different variations of SMOTE was made with a particular emphasis on the Geometric-SMOTE. These techniques were applied to LUCAS survey with the goal of improving the land cover classification of a supervised learning algorithm.

The most used algorithms to produce LCLU maps, such as Support Vector Machines, Random Forest, and Neural Networks like CNN's, are vector-based methodologies, which means that each pixel is seen as a data point in an order less feature space (Khatami et al., 2016). As a result, these approaches ignore the temporal correlation present in multi-temporal remote sensing data, which can be important to help distinguish some land cover classes (Ienco et al., 2017). The addition of temporal metrics can be used in the attempt to mitigate this issue of the temporal independence of the data points. These features can include statistical values, like percentiles, and some indexes, as maximum vegetation index (Pelletier et al., 2019). The usage of these type of metrics can be seen in Pflugmacher et al. (2019), where the addition of annual variance metrics had a very slightly improvement in the overall classification accuracy but had a more significant improvement in classes

like cropland, artificial land and snow and ice. Leinenkugel et al. (2019) also used band percentiles to exploit temporal patterns present in their multi-temporal dataset. In this study, all the different datasets used these temporal statistics, therefore no conclusion regarding their usefulness was made. In Pelletier et al. (2016), on the other hand, the addition of temporal metrics computed from NDVI did not yield any significant improvements on the classification results, but instead just increased the computational cost of the training process.

Most recently, with the developments in deep learning and in computer processing power, some studies used Recurrent Neural Network (RNN) to model the temporal correlation present in remote sensing time series, as this type of network can learn the patterns in sequential data. In the literature, two main RNN architectures have found success, Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU), as they are able to model long term sequences due to their gated system (Ma et al., 2019). This was the case in Rußwurn & Körner (2017), where a LSTM network was used to learn the temporal patterns present in land cover classes. This model outperformed other commonly used algorithms for LCLU classification like CNN and SVM, particularly in cropland classes, that have a more pronounced temporal behaviour. Jia et al. (2017) also found success in LCLU classification by applying a dual memory LSTM model, which contained two cell states, one to store long term variations and other to save short term changes. This approach was able not only to properly identify previously known land cover classes, but also to detect unseen classes. In contrast, Lyu et al. (2018) opted for a GRU network to detect long term urban changes in four different cities with similar climate conditions, to not only, reduce the impact of radiometric variations, but also to amplify the differences between classes. To avoid the issue of having a small number of pre-classified training data points, the authors used a transfer learning method that had two stages. Firstly, the model was trained only for one city and for a single year, and afterwards this model classified all the images used for this study and the points that add high likelihood scores were added to the train dataset to train the final model.

Some studies have compared both types of Recurrent Neural Networks (LSTM and GRU) to extract temporal patterns present in remote sensing images. Ndikumana et al. (2018) achieved better results with GRU in agriculture land cover classification with Sentinel-1 SAR images. This model proved to be very good in the classification of rice, which has a very pronounced temporal behaviour, achieving a F-measure value of 96% for that class. Rußwurm & Körner (2018) also found more success with GRU than with LSTM in crop classification. For this study, the authors train the algorithm with images that not only were not atmospherically corrected, but also add the presence of clouds. However, the network was able to consider it as noise, so there was no need to pre-classify clouds. Having said this, the algorithm was able to learn crop classification and to filter out clouds. In opposite direction, Mou et al. (2019) reach more success when applying a LSTM network for change detection. Although the LSTM only perform slightly better than the GRU, both models outperform other standard remote sensing classifiers, like SVM and CNN. The model presented in this study was composed by a convolutional layer, to extract spectral-spatial features from the images, a LSTM layer, to model the temporal dependencies and two fully connected layers in the end to perform the classification.

Recently, with the development of space-born Earth observation sensors, it became easier to have access to multi-spectral images with low time span and at no cost. Following this, alongside with technological improvements like, the increase of data storage capability and processing power, and the advances in machine learning, it became easier to take advantage of the temporal patterns

present in remote sensing data (Rußwurm & Körner, 2018). The Sentinel-2 mission, launched by ESA, is one good example of the developments in Earth observation, as it provides images with 13 spectral bands, which include the visible spectrum, near-infrared and short-wave infrared, at a spatial resolution ranging from 10 m to 60 m, depending on the band in question (Drusch et al., 2012). The Sentinel-2 Multispectral Imager (MSI) is a constellation of two satellites, Sentinel-2A and Sentinel-2B, in a sun-synchronous polar orbit, that became operational in 28 of November 2015 and 7 of July 2017, respectively (Liu et al., 2017). Each satellite takes 10 days to complete a rotation around the globe, which allows the MSI to have a 5-day revisit period (Pahlevan et al., 2017). Like this, Sentinel-2 has gained the interest of the remote sensing community as it provides images around the globe with high spatial resolution, containing a significant number of spectral bands, having a low time interval between images, which presents good opportunities to build time series, and being free of charge (Close et al., 2018).

In Weigand et al. (2020), the author opted to use Sentinel-2 images over the ones produced by its North American counterpart, Landsat, due to Sentinel's higher spatial resolution (10 m compared to Landsat's 30 m resolution), to reduce positional errors which are prone to happen when using the LUCAS survey as the reference data source. Topaloğlu et al. (2016) made a comparison between Sentinel-2 and Landsat-8 images for LCLU classification, having attained more success with Sentinel's images on both classifiers tested. In the end of the study, despite of the spatial resampling applied to the Sentinel's images, that converted them into 30 m spatial resolution images, the author concluded that the higher spatial resolution from Sentinel-2 had contributed for the gap observed between the results.

This literature review helped us identify the research lines for this study and guided us in the development of the methodology described in the next section.

3. METHODOLOGY

This section describes the approach taken in this study to the following subjects: the study area, the treatment made to the reference data and LCLU nomenclature chosen, the preprocessing applied to the data, the implementation of the algorithms and the evaluation procedure.

3.1. STUDY AREA

The chosen study area was the entire territory of Continental Portugal (Figure 1). This region is characterized by a great diversity of land cover, having the presence of Mediterranean and Atlantic landscapes. Most of the territory of Continental Portugal is occupied by forest, agriculture, and agroforests land, around 92%, and having only 5% of the area covered by artificial land.

Figure 1. Map of the study area (Continental Portugal).

3.2. LUCAS REFERENCE DATA

In this study, we used the LUCAS dataset from 2018 as reference data to train our supervised algorithms. The LUCAS survey, as mentioned above (Section 2), is a database, conducted by the Eurostat, that gathers LCLU information of land points across the territory of the European Union (Pflugmacher et al., 2019). The LCLU class is assigned to each point through photointerpretation at first instance, and then physically by a surveyor in a field visit (Griffiths et al., 2019). This survey distinguishes the points by 8 major LCLU classes, which then can be divided in 84 subclasses (Leinenkugel et al., 2019).

We decided to define a LCLU nomenclature that would allow for the creation of a meaningful LCLU map. To this end, we decided to go with a classic nomenclature aligned not only with the specificities of this study but also in line with the nomenclatures used in other studies developed by DGT. The established nomenclature had two levels of granularity. The first one was composed by 6 classes and it was very similar to the set of 8 major classes distinguished in the LUCAS, with the difference being the grouping of Shrubland with Natural herbaceous (Grassland) and Water with Wetlands. The second level of our nomenclature is more detailed, being composed by 12 LCLU classes, and it will be our main set of LCLU classes (Table 1). To properly apply the nomenclature defined, a correspondence between these LCLU classes and the LUCAS classes was established. To perform this, we tried to include all the LUCAS classes, present in the territory of Continental Portugal, into one of the LCLU classes defined in our nomenclature (Table 2). However, some LUCAS classes present in Portugal mainland were excluded from our study, whether because they did not fit into any class of our nomenclature, or because they could be bewildered between two or more LCLU classes, as it was the case for Olive groves, which dependent on the way that are planted and the maturity of the trees it can be classified as Broadleaved or as Agriculture.

Table 1. LCLU Nomenclature (Level-1 and Level-2)

LCLU map Level-1	LCLU map Level-2
Urban	Urban
Agriculture	Rainfed
	Irrigated
	Rice
	Pastures
Natural herbaceous and Shrubland	Natural herbaceous
	Shrubland
Forest	Broadleaved
	Coniferous
Bare soil	Bare soil
Water and Wetlands	Wetlands
	Water

Table 2. LCLU Level-2 classes with the correspondence to the LUCAS classes and their distribution

LCLU map Level-2	LUCAS class	Number of data points
Urban	A11, A12	85
Rainfed	B11, B12, B13, B14, B15, B18, B52, B54	165
Irrigated	B16, B21, B31, B41, B41	57
Rice	B17	19
Temporary pastures	B55, E10, E20 with LU class U111	780
Natural herbaceous	E10, E20 with LU class U415 or U420, E30	222
Shrubland	D10, D20	563
Broadleaved	C10	2304
Coniferous	C21, C22, C23	601
Bare soil permanent	F10, F20	37
Water	G11, G21	62
Wetlands	H11, H21	15

Since our study area was Continental Portugal, we only used the LUCAS sampling units within this territory (Figure 2). Furthermore, we applied some filters to get only the most reliable LUCAS sampling units to train our algorithms. So, we only selected the data points that were observed directly on point, which had more than 0.1 hectares of area, had more than 20 meters of width and where the LC class covered more than 50% of the area. In addition to these filters, we also excluded specific cases, keeping only the points with special remarks as “Harvested field”, “Tilled/sowed” or “No remark”, excluding Broadleaved points with the “Harvested filed” as the special remark, since it is similar to a “Clear cut”, which was excluded previously, and points classified as Spontaneously re-vegetated with the special remark of “Harvested field” or “Tilled/sowed”, as this case can be confused with bare soil.

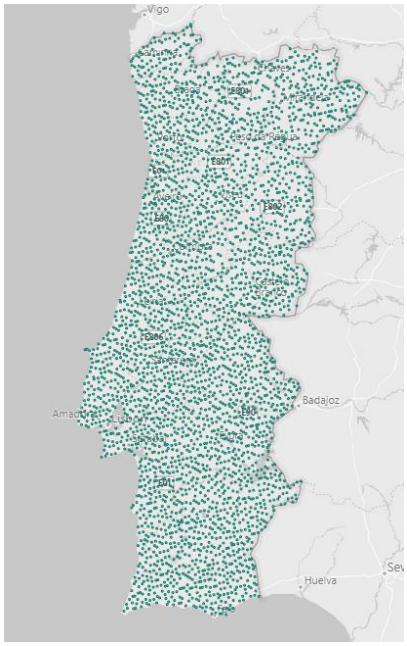

Figure 2. Map of the study area and LUCAS 2018 sample points.

It is important to notice, that our study area contains 7168 LUCAS points. However, with the filters applied, to consider only the most reliable sampling units and to remove the unwanted classes, the number of training points was reduced to 4910. Considering that this represents a big reduction on the number of sampling units available to train the algorithms (a reduction of more than 30%), and 1708 of these points (23,8% of the total number of points) are due to the filters related with the trustworthiness of the points, we decided to make a test to understand if the training of the algorithms would be better with a larger sample containing less reliable points or with a smaller sample only composed by trustworthy data points. To perform this experiment, we trained the algorithms with both samples, and the results showed that it was preferable to remove the less reliable sampling units. This test was made on the early stages of our study, and we decided to exclude the details from this document.

3.3. DATA PREPARATION

3.3.1. Input Data Normalization

To standardize the input data to train a machine learning model there are two main techniques: the z-norm and the min-max normalization. The z-norm transforms each input value by subtracting the mean and dividing it by the standard deviation of the input variable. This transformation makes all the values of the variable range between -1 and 1, having 0 as the mean and 1 as the standard deviation. On the other hand, the min-max normalization subtracts to all input values the minimum value and then divides it by the range of that input feature, which is the minimum value minus the maximum value. However, in Pelletier et al. (2019) some drawbacks to both methods are presented, as the authors stated that the application of the z-norm would affect the general trend of the time series and the min-max could suffer with the presence of extreme values (outliers). To overcome this problem, they presented an additional normalization technique which consists in using the 2% and the 98% percentile instead of the minimum and the maximum values, respectively, in the min-max normalization, with the goal of making this technique more robust to outliers. This technique was named global feature min-max normalization by the authors, and we will refer to it as global feature normalization through the rest of this document.

Following the methodology of the paper mentioned above, we implemented a test to assess which of these three approaches was best suited to our problem. This test consisted in running all three algorithms chosen for our study with the dataset standardized using each of the three techniques mentioned above. This was made in the early stages of our study, and the results showed that the global feature normalization was the best normalization technique to implement in our problem, and therefore, it was applied during the rest of our study. To properly apply this normalization without having data leakage, the percentiles used to calculate the global feature normalization were drawn from the training set for the standardization of both train and test datasets. We decided not to include the details of this experiment on this document.

3.3.2. Input Features

Sentinel-2 intra-annual surface reflectance imagery produced at DGT was used in this study. The data is acquired between October 2017 and September 2018, corresponding to the 2018 agricultural year

period in Portugal. For our study, we took into consideration only 10 of the 13 spectral bands available in Sentinel-2's imagery, having dropped the bands 1 (coastal aerosol), 9 (water vapor) and 10 (SWIR - cirrus) since they were conceived for atmospheric correction of the other bands. Monthly composites are produced using the median pixel value. Gap values that resulted from cloud and shadow mask, due to atmospheric correction, were filled using linear interpolation in time from the closest months. To better identify non-linear relationships between the spectral bands, spectral indices were also processed, which are commonly used by the remote sensing community for supervised classification studies (Pelletier et al., 2019). Five spectral indices were selected: The Normalized Burn Ratio (NBR), The Normalized Difference Buildup Index (NDBI), The Normalized Difference Middle Infrared index (NDMIR), The Normalized Difference Vegetation Index (NDVI) and The Normalized Difference Water Index of MCFeeters (NDWIF). Additionally, percentiles for each spectral band and index, as well as some gaps between percentiles were calculated. This improves the information concerning the temporal distribution of the input features (Pflugmacheret al., 2019). More specifically, we included the 10^{th}, 25^{th}, 50^{th}, 75^{th} and 90^{th} percentile, as well as the difference between the 10^{th} and the 90^{th} and the difference between the 25^{th} and 75^{th} percentile.

3.3.3. Train Oversampling Methods

Looking at the distribution of labeled sampling units present in our training dataset, we can observe that there is a big disparity between the number of training data points available for each class (Table 2), which can bias our algorithms towards the majority classes (Sáez et al., 2016). To solve this problem of imbalanced learning, we decided to test three different techniques that will modify the training set and make it more balanced. These three methods are variants of the SMOTE, which is one of the most popular oversampling techniques used in machine learning. The first method used is the standard SMOTE method, which generates artificial data points along the lines that connect the k nearest neighbours on the minority class (Fernández-Navarro et al., 2011). The second method is the Borderline SMOTE, which is a similar technique to the standard SMOTE, however it generates the new synthetic data points near to the decision boundary of the minority class. This is obtained by creating new sampling units along the lines that connect the support vectors of the minority class and their k nearest neighbours (Ngueyn et al., 2009). The third and final method used is also a variant of SMOTE and is called Geometric-SMOTE. This technique generates new artificial data points on a flexible geometric region around the minority class, which is controlled by the choice of the hyperparameters (Douzas et al., 2019).

To test the oversampling techniques, we generated three different training datasets by applying each one of the oversampling methods to original LUCAS data points. All the techniques applied increased the number of data points of the minority classes up to 150, as this was the number that yielded the best results. After this, we used these new datasets to train our machine learning models to analyze the impact that each oversampling technique had in the algorithm's results. To implement the oversampling techniques, we used the Imbalanced-Learn and the Geometric-SMOTE libraries of Python. The choice of hyperparameters was made through a series of tests using different parameter combinations, and in the end, choosing the best combination for each method. For all the oversampling techniques, the number of nearest neighbors was set to 5. Apart from that, for the Geometric-SMOTE, we also specify the selection strategy as majority, as well as the truncation and deformation factor, which were established as 1 and 8, respectively.

3.4. CLASSIFICATION ALGORITHMS

The main objective of this study is to evaluate the Recurrent Neural Network (RNN) suitability to produce LCLU classification of satellite image time series, having as reference data the LUCAS survey sampling units. To do this, we compared the performance of a GRU network to the performance of two state-of-the-art algorithms for remote sensing classification, Random Forest (RF), and Support Vector Machine (SVM). To define the parameters of these three algorithms we ran some tests to find the most suitable parameter combination for this problem. For these tests, we applied a stratified 3-fold cross validation to the training data alone, with the aim of defining these combinations of parameters independently of the test data. The small number of folds is explained by the reduced number of sampling units of some LCLU classes (e.g., Wetlands and Rice), as if this number was bigger, some splits would only get 1 or 2 data points for these classes. The stratified option will make sure that each fold has a similar class distribution to the one present in the training data. In this study, we will only report the results attained by the best combination of parameters for each algorithm and we will omit the tests related with the parametrization definition.

As mentioned in Section 2.1.1, the Random Forest is an ensemble classifier which is composed by many individual decision trees that are created based on bootstrap samples of the training data, and each split node is determined based on a random subset of variables (Liaw & Weiner, 2002). Even though the RF algorithm is not very affected by the parametrization (Inglada et al., 2017), we defined the following two parameters: the number of trees as 500 and the number of randomly selected features to perform each split as the square root of the total number of variables. The Support Vector Machine is a supervised classifier that aims to find the optimal hyperplane that can completely divide the data points of each class, while maximizing the margin. To define this hyperplane, the SVM only considers the sampling units that are closer to the decision boundary of the class, also called support vectors (Fauvel et al., 2008), as introduced in Section 2.1.2. For our application of the SVM, we only specified the kernel type, having it defined as linear. Both algorithms, RF and SVM, were implemented using the Scikit Learn library of Python, and all the parameters not mentioned above were set as the default values defined in this library.

For the RNN model, we choose the GRU architecture because it is less complex than the LSTM, while keeping the gated structure that allows to the model to learn the long-term relationships present in the data (Ho Tong Minh et al., 2018). In addition, since we have a small training sample, the GRU network will be easier to train properly, as it has less parameters to learn than the LSTM (Mou et al., 2017). After our parameterization tests, our final network was composed by a GRU layer with 32 neurons and a SoftMax layer to produce the output. The activation function used was the ReLU, the optimizer chosen was the Adam with a 0.001 learning rate and the loss function selected was the Categorical Crossentropy. In addition, the dropout and the recurrent dropout were set to 0.1 and 0.05, respectively, and the kernel initializer was defined as Normal. This network was trained from 20 epochs with a batch size of 10. This model was implemented in Python using the Keras library with Tensorflow as back end.

3.5. PERFORMANCE ASSESSMENT

To properly evaluate the performance of our models, data points which were not included in the training process are needed to avoid bias in the results. Our first approach was to split the LUCAS survey sampling units into train and test data, by using a split of 80%/20%, respectively. However, this presented two issues: firstly, it would reduce the number of training data points of LCLU classes which were already poorly represented, and secondly, it would make same LCLU classes almost unrepresented in the test set (e.g., there would only be 3 sampling units of Wetlands in the test dataset). So, this not only would accentuate the imbalance learning problem already present in the training but would also create a similar issue on the testing. To overcome this, an independent dataset to assess the performance of our models was provided by DGT. This validation dataset had 600 data points that were evenly distributed by the LCLU classes present in our study. The sampling units were labeled by technicians with expertise in LCLU using orthophoto maps of 2018 with a 25cm pixel. The metric chosen to evaluate the models was the Overall Accuracy that is the most common measure to evaluate model performance for LCLU classification (Liu et al., 2007). The Overall Accuracy consists of the number of well classified sampling units over the total number of data points (Mou et al., 2017).

4. RESULTS AND DISCUSSION

In this section, we will present the results of this study and compare them to results achieved in other studies. The disclosed results are the overall accuracy relative to each combination between classification algorithm and oversample technique for both levels of nomenclature, Level 2 (Table 3) and Level 1 (Table 4). These values were achieved when training the algorithms with the entire LUCAS dataset (Section 3.2) and testing them with an independent dataset (Section 3.5) to keep the integrity of the results achieved. Our analysis will mainly focus the results achieved with Level 2 nomenclature and then we will compare these results to the ones attained with the Level 1 nomenclature.

Table 3. Matrix of the Overall Accuracy achieved by each algorithm combined with each oversampling technique using the Level 2 class nomenclature.

		Oversampling technique			
Algorithms		None	Standard	Borderline	Geometric
	RF	56.7	60.3	60.2	60.2
	SVM	61.7	61.2	62.7	62.2
	GRU	59.5	60.2	61.3	61.8

Table 4. Matrix of the Overall Accuracy achieved by each algorithm combined with each oversampling technique using the Level 1 class nomenclature.

		Oversampling technique			
Algorithms		None	Standard	Borderline	Geometric
	RF	71.3	73.8	74.0	74.2
	SVM	74.3	73.3	74.5	73.5
	GRU	73.0	73.3	74.7	74.2

Looking at the results attained with the Level 2 nomenclature, we can see that they do not have a big variance between then, as they range between 56.7% and 62.7%. Regarding the classifiers, the best performing algorithm was the Support Vector Machine, with higher accuracy results than the other classifiers independently of the oversampling technique used. On the other hand, the Random Forest algorithm was the worst performing classifier, even though it got the best result when combined with the standard SMOTE, as the difference between the scores achieve by the different algorithms is insignificant (0.1%). In the end, we can observe that the classifiers rank up in the same way, independently of the oversampling methods used, being the SVM the most suited model for this problem, followed by the GRU and finally the RF classifier.

A similar analysis cannot be made for the oversampling techniques since no approach is clearly better than all the others, as each algorithm obtained higher accuracies with a different oversample technique. However, we can notice that when not applying any oversampling strategy generally provides worst results, apart from the SVM. Not applying oversampling to the training data produces

significantly lower results for the RF and is also the lowest result for the GRU. Additionally, we observe that both Borderline SMOTE and Geometric SMOTE get better results that their standard counterpart, since the standard SMOTE is only better than these approaches when combined with the RF algorithm and the difference between the accuracy scores is negligible. Finally, it is worth to notice that the Borderline SMOTE obtains its best result with the SVM, which could be explained by both techniques being centred around the decision boundary that splits the different classes, and the Geometric SMOTE finds more success with the GRU.

Comparing the results achieved when using the Level 1 nomenclature with the ones attained with the Level 2 nomenclature, we can see that the values of the accuracy increased, which is expected considering the smaller number of classes in the Level 1 nomenclature. Focusing our analysis on the oversampling techniques, we can see a similar behaviour between the two nomenclatures, since the worst option continues to be not applying any type of oversampling (with the same exception in the SVM). Borderline SMOTE and Geometric SMOTE remain with higher accuracies when compared with their standard counterpart. On the other hand, when analysing the performance of the classifiers, we can observe a slight shift on the results, as the GRU not only achieved the best result with this nomenclature (when parred with the Borderline SMOTE), but also improved his performance in comparison with the SVM. The RF also attained better results when compared to the ones achieved by the other classifiers, making the SVM the classifier that improved less with the reduction of the number of LC classes.

Looking to the results attained in other studies that also used LUCAS survey to train a supervised classifier, we can see that in Close et al. (2018), the authors reached much higher accuracy scores then the ones presented in our study using standard machine learning algorithms. This paper was able to achieve an accuracy of 91.1% using a Maximum Likelihood classifier trained only using LUCAS sampling units and tested with an independent dataset, as in our study. However, this paper only took into consideration the Belgium territory and 5 LC classes, a much lower number of classes when compared to our 12-class nomenclature. Pflugmacher et al. (2019) used LUCAS to train a RF classifier to make LC classification for the entire Europian territory, and was able to attained an higher accuracy than the one achieve in our study (75.1%). This article uses the same number of LCLU classes as our study (12 classes), but with a slight diference in the nomenclature chosen, as this aritcle includes the classes Snow/ice and Mixed forest while our nomenclature includes Rice and Pastures. However, the fact that this article had as its study area the entire territory of Europe, it was able to leverage on entire LUCAS survey to train the classifier, which could have helped the classification of LC classes with less representativeness. Aditionally, this paper applied cross validation to the LUCAS to achieve their results, therefore not using an independent dataset to attaine them.

Leinenkugel et al. (2019) made a classification of three different study areas being one of them Continental Portugal. This paper used different open source datasets to train a RF to make LCLU classification. The LCLU nomenclature chosen in this article was very similar to the one used in our study, not including the classes Rice and Pasture and adding the Mixed Forest. If we look to the results of this article, we can observe that the best accuracy attained for Portugal was 59.2%, which is slightly worse than our best result (62.7%). However, this value was acheived when the RF was trained with all the datasets available in this paper. When the training of the algorithm only had the LUCAS dataset as reference data, the accuracy attained dropped significantly to 44.3%, a value much

lower than the one achieved in our study. It is also important to notice, that the results attained for the Portugal area are always lower than the other two regions contemplated in this study, which could be due to Portugal being a region characterized by a great diversity of soil occupation. Another paper that used LUCAS as reference data and had Portugal as study area was Douzas et al. (2019), even though it only considered part of the north-western territory of Portugal and it only considered the 8 major classes of LUCAS survey. This study focused more on the oversampling techniques used to increase the performance of the machine learning algorithms. In the end, it concluded that the Geometric SMOTE out perfomed the other techiniques used, which included the standard SMOTE and the Borderline SMOTE among others. The same conclusion can not be taken in our study, since the Borderline SMOTE out perfomed the Geometric SMOTE when the SVM was used as classifier.

5. CONCLUSIONS

In this study, we applied a GRU network to make LCLU classification of a satellite image time series over the territory of Continental Portugal. To train this model, we used the freely accessible LUCAS survey as our reference data, and we used the Sentinel-2 images as our data source. To assess the performance of our GRU, we also trained other two state-of-the-art classifiers used for remote sensing to compare it with (Random Forest and Support Vector Machine). Since LUCAS survey is a very unbalanced dataset, with some classes with very low number of sampling units, three oversampling techniques were applied to alleviate these limitations (SMOTE, Borderline SMOTE and Geometric SMOTE).

Looking at the results, it is shown that the GRU network did not outperformed the state-of-the-art classifiers when trained with a limited number of training sampling units, as it performed worse than the SVM classifier and it just performed slightly better than the RF. This could be explained by two main factors: the addition of new input features that can give information about the temporal behavior of the different LCLU classes to the standard machine learning methods used in this study; and the small number of training data points when compared to the number of LCLU classes, which affects the training of neural networks in general, especially deeper networks. Furthermore, we can conclude that applying oversampling to the LUCAS survey to increase the number of sampling units of the minority classes and reduce the unbalance of the training set, improves the overall results. Additionally, we can see that the application of the variants of SMOTE (Borderline and Geometric) achieve slightly better results than their standard counterpart. Finally, this study shows that we can successfully use Sentinel-2 imagery and LUCAS survey data to train machine learning models to produce LCLU classification of satellite image time series, which can leverage many other studies on this topic since both sources of data are freely accessible.

6. LIMITATIONS AND RECOMMENDATIONS FOR FUTURE WORKS

The biggest limitation of this study was related with the LUCAS survey itself, as this dataset not only contained a small number of sampling units, but also, was severely unbalanced, which made the training of the machine learning algorithms a challenging task. The other limitation faced in this study was related with the computational power needed to train and fine tune the models. To overcome this limitation, we used Google Colab to have access to a GPU and more processing power.

In future works, similar experiences should be conducted having a different study area, maybe considering a country/region, not only with a larger area, but also less complex in terms of land cover, as this could increase the number of LUCAS sampling units available and reduce discrepancies between the number of training points for each class. Some studies could also compare the performance of the models mentioned in this study, but without adding new input features, to understand the impact of these type variables, that aim to give temporal information about the LCLU classes. Finally, it would be interesting to also explore the 3 domains of satellite images (spatial, spectral, and temporal), developing a Convolutional Recurrent Neural Network that would be fed by multispectral information, and so, creating a LCLU classification model that would be trained using the LUCAS survey as the reference data.

7. BIBLIOGRAPHY

Agency, E. E. (2020, 06 29). *CORINE Land Cover*. Retrieved from Copernicus: https://land.copernicus.eu/pan-european/corine-land-cover

Chen, Y., Lin, Z., Zhao, X., Wang, G., & Gu, Y. (2014). Deep Learning-Based Classification of Hyperspectral Data. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 2094-2107.

Close, O., Benjamin, B., Petit, S., Fripiat, X., & Hallot, E. (2018). Use of Sentinel-2 and LUCAS Database for Inventory of Land Use, Land Use Change, and Forestry in Wallonia, Belgium. *Land*.

Deilman, B. R., Ahmad, B. B., & Zabihi, H. (2014). Comparison of two classification methods (MLC and SVM) to extract land use and land cover in Johor Malaysia. *IOP Conf. Series: Earth and Environmental Science*.

Douzas, G., Bacao, F., Fonseca, J., & Khudinyan, M. (2019). Imbalanced Learning in Land Cover Classification: Improving Minority Classes' Prediction Accuracy Using the Geometric SMOTE Algorithm. *Remote Sensing*.

Drusch, M., Del Bello, U., Carlier, S., Colin, O., Fernandez, V., Gascon, F., . . . Bargellini, P. (2012). Sentinel-2: ESA's Optical High-Resolution Mission for GMES Operational Services. *Remote Sensing of Environment*.

Fauvel, M., Benediktsson, J. A., Chanussot, J., & Sveinsson, J. R. (2008). Spectral and Spatial Classification of Hyperspectral Data Using SVMs and Morphological Profiles. *IEEE Transactions on Geoscience and Remote Sensing*.

Fernández-Navarro, F., Hervás-Martínez, C., & Gutiérrez, P. A. (2011). A dynamic over-sampling procedure based on sensitivity for multi-class problems. *Pattern Recognition*, 1821-1833.

Flamary, R., Fauvel, M., Dalla Mura, M., & Valero, S. (2015). Analysis of Multitemporal Classification Techniques for Forecasting Image Time Series. *IEEE Geoscience and Remote Sensing Letters*.

Gers, F. A., Schmidhuber, J., & Cummins, F. (1999). Learning to Forget: Continual Prediction with LSTM. *1999 Ninth International Conference on Artificial Neural Networks ICANN 99.* Edinburgh, UK: IET.

Griffiths, P., Nendel, C., & Hostert, P. (2019). Intra-annual reflectance composites from Sentinel-2 and Landsat for national-scale crop and land cover mapping. *Remote Sensing of Environment*, 135-151.

Heine, I., Jaghuber, T., & Itzerott, S. (2016). Classification and Monitoring of Reed Belts Using Dual-Polarimetric TerraSAR-X Time Series. *Remote Sensing*.

Helber, P., Bischke, B., Dengel, A., & Borth, D. (2019). EuroSAT: A Novel Dataset and Deep Learning Benchmark for Land Use and Land Cover Classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 2217-2226.

Ho Tong Minh, D., Ienco, D., Gaetano, R., Lalande, N., Ndikumana, E., Osman, F., & Maurel, P. (2018). Deep Recurrent Neural Networks for Winter Vegetation Quality Mapping via Multitemporal SAR Sentinel-1. *IEEE Geoscience and Remote Sensing Letters*.

Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. *Neural Computation 9*, 1735-1780.

Huang, B., Zhao, B., & Song, Y. (2018). Urban land-use mapping using a deep convolutional neural network with high spatial resolution multispectral remote sensing imagery. *Remote Sensing of Environment*, 73-86.

Ienco, D., Gaetano, R., Dupaquier, C., & Maurel, P. (2017). Land Cover Classification via Multi-temporal Spatial Data by Recurrent Neural Networks. *IEEE Geoscience and Remote Sensing Letters*, 1685-1689.

Inglada, J., Vincent, A., Arias, M., Tardy, B., Morin, D., & Rodes, I. (2017). Operational High Resolution Land Cover Map Production at the Country Scale Using Satellite Image Time Series. *Remote Sensing*.

Jia, X., Khandelwal, A., Nayak, G., Gerber, J., Carlson, K., West, P., & Kumar, V. (2017). Incremental Dual-memory LSTM in Land Cover Prediction. *The 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, (pp. 867-876).

Khatami, R., Mountrakis, G., & Stehman, S. V. (2016). A meta-analysis of remote sensing research on supervised pixel-based land-cover image classification processes: General guidelines for practitioners and future research. *Remote Sensing of Environment*, 89-100.

Krawczyk, B., Cano, A., & Wozniak, M. (2018). Selecting local ensembles for multi-class imbalanced data classification. *International Joint Conference on Neural Networks.*

Leinenkugel, P., Deck, R., Huth, J., Ottinger, M., & Mack, B. (2019). The Potential of Open Geodata for Automated Large-Scale Land Use and Land Cover Classification. *Remote Sensing*.

Liaw, A., & Weiner, M. (2002). Classification and Regression by RandomForest. *R News*, 18-22.

Litjens, G., Kooi, T., Bejnordi, B. E., Setio, A. A., Ciompi, F., Ghafoorian, M., . . . Sánchez, C. I. (2017). A Survey on Deep Learning in Medical Image Analysis. In *Medical Image Analysis* (pp. 60-88).

Liu, C., Frazier, P., & Kumar, L. (2007). Comparative assessment of the measures of thematic classification accuracy. *Remote Sensing of Environment*.

Liu, H., Li, Q., Shi, T., Hu, S., Wu, G., & Zhou, Q. (2017). Application of Sentinel 2 MSI Images to Retrieve Suspended Particulate Matter Concentrations in Poyang Lake. *Remote Sensing*.

Lyu, H., Lu, H., & Mou, L. (2016). Learning a Transferable Change Rule from a Recurrent Neural Network for Land Cover Change Detection. *Remote Sensing*.

Lyu, H., Lu, H., Mou, L., Li, W., Wright, J., Li, X., . . . Gong, P. (2018). Long-Term Annual Mapping of Four Cities on Different Continents by Applying a Deep Information Learning Method to Landsat Data. *Remote Sensing*.

Ma, L., Liu, Y., Zhang, X., Ye, Y., Yin, G., & Johnson, B. A. (2019). Deep Learning in remote sensing applications: A meta-analysis and review. *ISPRS Journal of Photogrammetry and Remote Sensing*, 166-177.

Mack, B., Leinenkugel, P., Kuenzer, C., & Dech, S. (2017). A semi-automated approach for the generation of a new land use and land cover product for Germany based on Landsat time-series and Lucas in-situ data. *Remote Sensing Letters*, 244-253.

Mack, B., Leinenkugel, P., Kuenzer, C., & Dech, S. (2017). A semi-automated approach for the generation of a new land use and land cover product for Germany based on Landsat time-series and LUcas in-situ data. *Remote Sensing Letters*, 244-253.

Melgani, F., & Bruzzone, L. (2004). Classification of Hyperspectral Remote Sensing Images With Support Vector Machines. *IEEE Transactions on Geoscience and Remote Sensing*.

Mou, L., Bruzzone, L., & Zhu, X. X. (2019). Learning Spectral-Spatial-Temporal Features via a Recurrent Convolutional Neural Network for Change Detection in Multispectral Imagery. *IEEE Transactions on Geoscience and Remote Sensing*, 924-935.

Mou, L., Ghamisi, P., & Zhu, X. X. (2017). Deep Recurrent Neural Networks for Hyperspectral Image Classification. *IEEE Transactions od Geoscience and Remote Sensing*.

Mountrakis, G., Im, J., & Ogole, C. (2011). Support vector machines in remote sensing: A review. *ISPRS Journal of Photogrammetry and Remote Sensing*, 247-259.

Ndikumana, E., Ho Tong Minh, D., Baghdadi, N., Courault, D., & Hossard, L. (2018). Deep Recurrent Neural Network for Agricultural Classification using multitemporal SAR Sentinel-1 for Camargue, France. *Remote Sensing*.

Ngueyn, H. M., Cooper, E. W., & Kamei, K. (2009). Borderline Over-sampling for Imbalanced Data Classification. *Fifth International Workshop on Computational Intelligence & Applications*, (pp. 24-29). Japan.

Pahlevan, N., Franz, B., Balasubramanian, S., & He, J. (2017). Sentinel-2 MultiSpectral Instrument (MSI) data processing for aquatic science applications: Demonstrations and validations. *Remote Sensing of Environment*, 47-56.

Pan, B., Shi, Z., & Xu, X. (2017). MugNet: Deep Learning for Hyperspectral Image Classification Using Limited Samples. *ISPRS Journal of Photogrammetry and Remote Sensing*.

Pelletier, C., Valero, S., Inglada, J., Champion, N., & Dedieu, G. (2016). Assessing the robustness of Random Forests to map land cover with high resolution satellite image time series over large areas. *Remote Sensing of Environment*, 156-168.

Pelletier, C., Valero, S., Inglada, J., Champion, N., Sicre, C. M., & Dedieu, G. (2017). Effect of Training Class Label Noise on Classification Performances for Land Cover Mapping with Satellite Image Time Series. *Remote Sensing*.

Pelletier, C., Webb, G. I., & Petitjean, F. (2019). Temporal Convolutional Neural Network for the Classification of Satellite Image Time Series. *Remote Sensing*.

Pflugmacher, D., Rabe, A., Peters, M., & Hostert, P. (2019). Mapping pan-European land cover using Landsat spectral-temporal metrics and the European LUCAS survey. *Remote Sensing of Environment*, 583-595.

Rußwurm, M., & Körner, M. (2018). Multi-Temporal Land Cover Classification with Sequential Recurrent Encoders. *ISPRS International Journal of Geo-Information*.

Rußwurn, M., & Körner, M. (2017). Temporal Vegetation Modelling using Long Short-Term Memory Networks for Crop Identification from Medium-Resolution Multi-Spectral Satellite Images. *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops.* Honolulu: IEEE.

Sáez, J. A., Krawczyk, B., & Wozniak, M. (2016). Analysing the oversampling of different classes and types of examples in multi-class imbalanced datasets. *Pattern Recognition*, 164-178.

Storie, C. D., & Henry, C. J. (2018). Deep Learning Neural Networks for Land Use Land Cover Mapping. *IEEE*, 3445-3448.

Sutskever, I., Martens, J., & Hinton, G. (2011). Generating Text with Recurrent Neural Networks. *Proceedings of the 28th International Conference on Machine Learning.* Bellevue, Washington, USA.

Taati, A., Sarmadian, F., Mousavi, A., Pour, C. T., & Shahir, A. H. (2015). Land Use Classification using Support Vector Machine and Maximum Likelihood Algorithms by Landsat 5 TM images. *Walailak Journal of Science and Technology*.

Topaloğlu, R. H., Sertel, E., & Musaoğlu, N. (2016). Assessment of Classification Accuracies of Sentinel-2 and Landsat-8 data for Land Cover/Use Mapping. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*.

Weigand, M., Staab, J., Wurm, M., & Taubenböck, H. (2020). Spatial and semantic effects of LUCAS samples on fully automated land use/ land cover classification in high-resolution Sentinel-2 data. *International Journal of Applied Earth Observation and Geoinformation*.

Xia, J., Chanussot, J., Du, P., & He, X. (2015). Rotation-based Support Vector Machine Ensemble in Classification of Hyperspectral data with limited training samples. *IEEE Transactions on Geoscience and Remote Sensing*.

Zhang, C., Sargent, I., Pan, X., Li, H., Gardiner, A., Hare, J., & Atkinson, P. M. (2019). Joint Deep Learning for land cover and land use classification. *Remote Sensing of Environment*, 173-187.

Zhuo, L., Zheng, J., Wang, F., Li, X., Ai, B., & Qian, J. (2008). A Genetic Algorithm Based Wrapper Feature Selection Method for Classification of Hyperspectral Images Using Support Vector Machine. *Proceedings of SPIE - The International Society for Optical Engineering*.

8. APPENDIX

Land cover code	Land cover	Observation
8	Not relevant	Point (1.5m radius)
A11	Buildings with 1 to 3 floors	Point (1.5m radius)
A12	Buildings with more than 3 floors	Point (1.5m radius)
A13	Greenhouses	Point (1.5m radius)
A21	Non built-up area features	Point (1.5m radius)
A22	Non built-up linear features	Point (1.5m radius)
A30	Other artificial areas	Point (1.5m radius)
B11	Common wheat	Point (1.5m radius)
B12	Durum wheat	Point (1.5m radius)
B13	Barley	Point (1.5m radius)
B14	Rye	Point (1.5m radius)
B15	Oats	Point (1.5m radius)
B16	Maize	Point (1.5m radius)
B17	Rice	Point (1.5m radius)
B18	Triticale	Point (1.5m radius)
B19	Other cereals	Point (1.5m radius)
B21	Potatoes	Point (1.5m radius)
B22	Sugar beet	Point (1.5m radius)
B23	Other root crops	Point (1.5m radius)
B31	Sunflower	Point (1.5m radius)
B32	Rape and turnip rape	Point (1.5m radius)
B33	Soya	Point (1.5m radius)
B34	Cotton	Point (1.5m radius)
B35	Other fibre and oleaginous crops	Point (1.5m radius)
B36	Tobacco	Point (1.5m radius)
B37	Other non-permanent industrial crops	Point (1.5m radius)
B41	Dry pulses	Point (1.5m radius)
B42	Tomatoes	Point (1.5m radius)
B43	Other fresh vegetables	Point (1.5m radius)
B44	Floriculture and ornamental plants	Point (1.5m radius)
B45	Strawberries	Point (1.5m radius)
B51	Clovers	Point (1.5m radius)
B52	Lucerne	Point (1.5m radius)
B53	Other leguminous and mixtures for fodder	Point (1.5m radius)
B54	Mixed cereals for fodder	Point (1.5m radius)
B55	Temporary grasslands	Point (1.5m radius)
B71	Apple fruit	Extended window (20m radius)
B72	Pear fruit	Extended window (20m radius)
B73	Cherry fruit	Extended window (20m radius)
B74	Nuts trees	Extended window (20m radius)
B75	Other fruit trees and berries	Extended window (20m radius)
B76	Oranges	Extended window (20m radius)
B77	Other citrus fruit	Extended window (20m radius)
B81	Olive groves	Extended window (20m radius)

B82	Vineyards	Extended window (20m radius)
B83	Nurseries	Extended window (20m radius)
B84	Permanent industrial crops	Extended window (20m radius)
Bx1	Arable land (only PI)	Point (1.5m radius)
Bx2	Permanent crops (only PI)	Extended window (20m radius)
C10	Broadleaved woodland	Extended window (20m radius)
C21	Spruce dominated coniferous woodland	Extended window (20m radius)
C22	Pine dominated coniferous woodland	Extended window (20m radius)
C23	Other coniferous woodland	Extended window (20m radius)
C31	Spruce dominated mixed woodland	Extended window (20m radius)
C32	Pine dominated mixed woodland	Extended window (20m radius)
C33	Other mixed woodland	Extended window (20m radius)
D10	Shrubland with sparse tree cover	Extended window (20m radius)
D20	Shrubland without tree cover	Extended window (20m radius)
E10	Grassland with sparse tree/shrub cover	Extended window (20m radius)
E20	Grassland without tree/shrub cover	Extended window (20m radius)
E30	Spontaneously vegetated surfaces	Extended window (20m radius)
F10	Rocks and stones	Extended window (20m radius)
F20	Sand	Extended window (20m radius)
F30	Lichens and moss	Extended window (20m radius)
F40	Other bare soil	Extended window (20m radius)
G11	Inland fresh water bodies	Point (1.5m radius)
G12	Inland salty water bodies	Point (1.5m radius)
G21	Inland fresh running water	Point (1.5m radius)
G22	Inland salty running water	Point (1.5m radius)
G30	Transitional water bodies	Point (1.5m radius)
G40	Marine sea	Point (1.5m radius)
G50	Glaciers, permanent snow	Point (1.5m radius)
H11	Inland marshes	Extended window (20m radius)
H12	Peatbogs	Extended window (20m radius)
H21	Salt marshes	Extended window (20m radius)
H22	Salines and other chemical deposits	Extended window (20m radius)
H23	Intertidal flats	Extended window (20m radius)